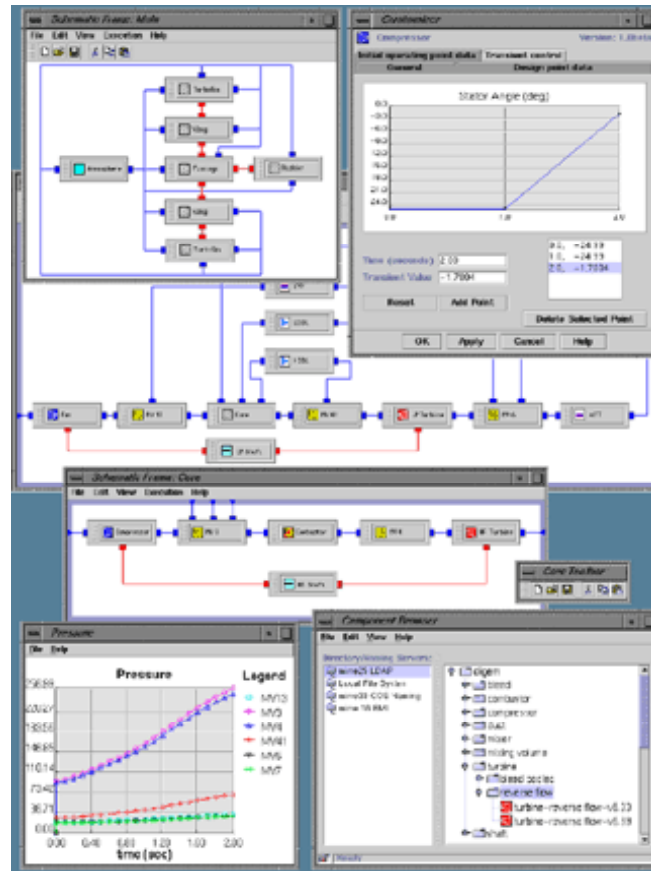


# Onyx-Advanced Aeropropulsion Simulation Framework Created

The Numerical Propulsion System Simulation (NPSS) project at the NASA Glenn Research Center is developing a new software environment for analyzing and designing aircraft engines and, eventually, space transportation systems. Its purpose is to dramatically reduce the time, effort, and expense necessary to design and test jet engines by creating sophisticated computer simulations of an aerospace object or system (refs. 1 and 2). Through a university grant as part of that effort, researchers at the University of Toledo have developed *Onyx*, an extensible Java-based (Sun Micro-systems, Inc.), object-oriented simulation framework, to investigate how advanced software design techniques can be successfully applied to aeropropulsion system simulation (refs. 3 and 4).

The design of Onyx's architecture enables users to customize and extend the framework to add new functionality or adapt simulation behavior as required. It exploits object-oriented technologies, such as design patterns, domain frameworks, and software components, to develop a modular system in which users can dynamically replace components with others having different functionality.

The accompanying figure shows a sample simulation session developed using Onyx's configurable Visual Assembly interface. Icons, representing available aerospace component models, are selected from a network repository and dragged into a model editor window. The icons are then interconnected to form a schematic diagram of the overall model. Customizing forms are provided for entering or editing data for each component, as well as for defining and controlling the numerical methods used to execute the simulation. Plotting capabilities, help browsers, text editors, and other user interface utilities are also included in the framework.



*Onyx Visual Assembly interface showing an example simulation model.*

The component models represented in the Visual Assembly interface are defined by Onyx's Common Engineering Model (CEM). CEM is a hierarchical object model that forms the foundation for an integrated representation of an aerospace system, its components, subcomponents, and sub-assemblies. Using CEM, users can create primitive component models, and visually or programmatically combine them to form more complex models. In addition, CEM can accommodate models having differing fidelity and discipline. Higher order analysis methods, such as computational fluid dynamics or finite element analysis, can be integrated within a component object with a small amount of programming. This feature enables a more concurrent engineering approach by allowing users to select a particular analysis method based on the level of detail needed, the objective of the simulation, the available knowledge, and the given resources. A general mechanism is provided to handle data transformation when components of different disciplines and fidelities are connected.

The introduction of interdisciplinary models and models having differing levels of fidelity requires support for distributed computing as it cannot be assumed that the higher fidelity software will run efficiently (or at all) on the same computer platform as the rest of the system. Onyx supports distributed computing by using several common software distribution mechanisms, including Java's RMI (Remote Method Invocation) and CORBA (Common Object Request Broker Architecture). Using CORBA objects, one can integrate

within the Onyx framework legacy software packages that must operate on specific architectures or operating systems and include them in a simulation mode.

## References

1. Claus, R.W., et al.: Numerical Propulsion System Simulation. Comput. Syst. Eng., vol. 2, no. 4, 1991, pp. 357-364.
2. Evans, A.L, et al.: An Integrated Computing and Interdisciplinary Systems Approach to Aeropropulsion Simulation. ASME Paper 97-GT-303, 1998.
3. Reed, John A.; and Afjeh, Abdollah A.: Computational Simulation of Gas Turbines: Part I--Foundations of Component-Based Models. J. Eng. Gas Turbines Power, vol. 122, 2000, pp. 366-376.
4. Reed, John A.; and Afjeh, Abdollah A.: Computational Simulation of Gas Turbines: Part 2--Extensible Domain Framework. J. Eng. Gas Turbines Power, vol. 122, 2000, pp. 377-386.

### Glenn contact:

Gregory J. Follen, 216-433-5193, Gregory.J.Follen@grc.nasa.gov

**Author:** John A. Reed

**Headquarters program office:** OAT

**Programs/Projects:** HPCCP, CAS

**Special recognition:** Best Paper Award, 1999 ASME International Gas Turbine Institute, Aircraft Engine Committee